EXHIBIT 1
Part 4 of 4

of a phrase structure rule, thus resulting in a translation according to the corresponding translation rule. (Clark Decl. ¶ 87.)

Each next legal choice in the parse state is thus an "element," as described in the '526 patent, having both a generic command component and a corresponding command action value. (Clark Decl. ¶ 88.)

4. [10B.3] "the parser identifying one of the elements as a best match relative to the generic command; and"

As explained with reference to claim element 1C.3 above, the token recognizer subcomponent of the parser-translator matches successive input words received from the user with the potential next legal choices in the parse state. (Clark Decl. ¶¶ 59, 89; Ex. 1002 at 18:38-42, 18:66-19:3.) When the input word matches a next legal choice that completes a phrase (*e.g.*, a valid pathname matching the next legal choice "pathname-expert" in the rule specification RULE1 (Ex. 1002 at 15:50-62)), the parser identifies that element as a best match for the generic command. (Clark Decl. ¶ 89; Ex. 1002 at 20:45-48.)

5. [10C.1] "a plurality of translators configured for issuing commands for the management programs according to respective command formats,"

The parser-translator of Martinez-Guerra is configured for issuing commands for a management program according to respective command formats. Martinez-Guerra discloses that multiple implementations of the parser-translator are possible, including one in which "multiple instances of a parser-translator

component, each with a corresponding grammar, provide multiple software applications with a common interface to high-level user language statements." (Ex. 1002 at 3:43-46.) An implementation of multiple parser-translator components includes a "plurality of translators." (Clark Decl. ¶ 90.) Likewise, Martinez-Guerra describes an embodiment "wherein operation of [a] single parser-translator component is suitably defined for each software application using a corresponding grammar encoding" (Ex. 1002 at 3:51-53), which a POSA would understand is functionally equivalent to a plurality of translators. (Clark Decl. ¶ 90.)

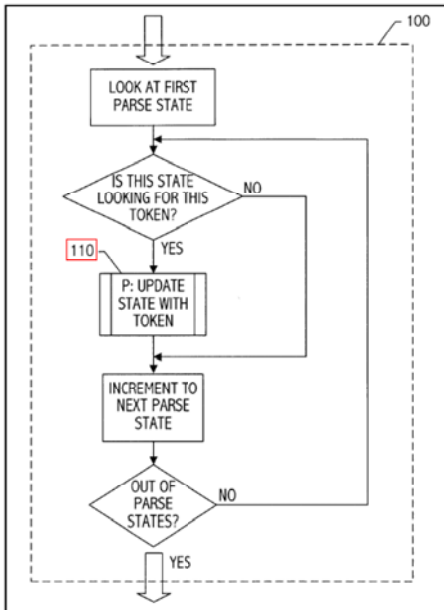
As Martinez-Guerra explains, the disclosed translators are configured for issuing commands for management programs according to respective command formats. This is why each of the instances of parser-translators has "a corresponding grammar." (Ex. 1002 at 3:53; *see also id.* at 3:43-53; Clark Decl. ¶ 91.) For example, Martinez-Guerra discloses that a translator subcomponent of the parser-translator issues an appropriate translation of the generic command "delete /usr/extract/testing" for the UNIX environment as "rm /usr/extract/testing." (Clark Decl. ¶ 91; Ex. 1002 at 15:59-16:5.) This is a respective command format of the UNIX environment because "rm" is a UNIX shell command to delete the target file. (Clark Decl. ¶ 91.)

6. [10C.2] "the parser outputting a prescribed command to a selected one of the translators based on the identified one element."

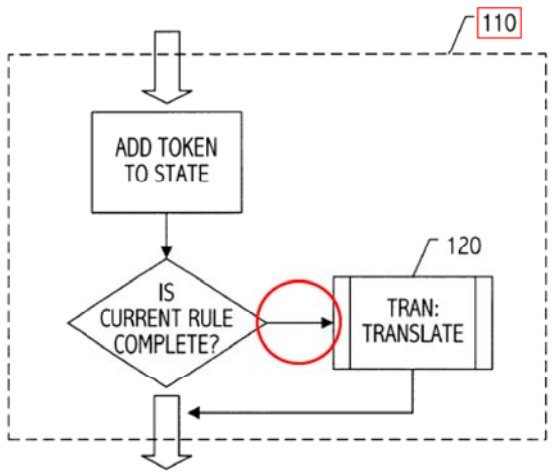
Martinez-Guerra discloses that, once a phrase structure rule is complete, which occurs when the final element of that rule is matched with a token from the input stream (Ex. 1002 at 20:45-48), the parser outputs a prescribed command to a translator based on that element. (Clark Decl. ¶ 92.) Figures 10 and 11 of Martinez-Guerra illustrate this process. Figure 10 discloses the process for determining whether an input token matches the next legal choice for one of the parse states maintained by the parser. (Clark Decl. ¶¶ 92; 20:40-42.) If the input token is found in the portion of the command parse tree containing that phrase structure rule, then the parser updates the parse state by adding the token using process 110 of Figure 10. (Clark Decl. ¶ 92; Ex. 1002 at 20:49-58.)

Process 110 is illustrated in Figure 11, wherein the parser adds the token to the parse state and determines whether that addition has completed a phrase structure rule (*e.g.*, in the example discussed above, the final token, "pathname-expert," has been received). (Clark Decl. ¶ 92; Ex. 1002 at 20:42-45.) If the token completes the rule, Figure 11 illustrates that the parser outputs a prescribed command, identified by the arrow directed to box 120, to a translator ("TRAN:"). (Clark Decl. ¶ 92; Ex. 1002 at 20:40-49.) Box 120, illustrated in Figure 12, is the

process undertaken by a translator subcomponent of the parser-translator. (Clark Decl. ¶ 92; Ex. 1002 at 20:45-48, Fig. 12.)



(Fig. 10 (annotation added).)



(Fig. 11 (annotation added).)

Because the command represented in Figure 11 is output by the parser once the final element of a phrase structure rule has been matched, it is output "based on the identified one element." (Clark Decl. ¶ 93.)

7. [11A.1] "The system of claim 10, wherein the parser further comprises a command word translation table configured for storing for each prescribed command word a corresponding token for identification of a matching token,"

Martinez-Guerra discloses that the "parser-translator 10 receives an input stream 6 from which token recognizer 11 extracts tokens in accordance with dictionary entries and rules encoded in the grammar 17 and in accordance with the

current parse state." (Ex. 1002 at 9:35-40.) As illustrated in Figures 1-3, the grammar—which Martinez-Guerra explains includes the dictionary entries—is depicted as a separate functional unit from the parser-translator. (*Id.* at Figs. 1-3, 14:8-18; Clark Decl. ¶ 94.) The "dictionary entries" used by the parser-translator "describe all the tokens that a parser-translator . . . should recognize," (Ex. 1002 at 14:16-18) and as explained above in relation to claim element 2B, parallel the '526 patent's description of the command word translation table ("includ[ing] all the command words 26 that are valid according to the generic syntax" (Ex. 1001 at 3:44-46)).

One of ordinary skill in the art would understand from these disclosures, and in view of Martinez-Guerra's example of the "delete /usr/extract/testing" generic command, that the dictionary entries are a command word translation table coextensive with the operation of the parser-translator. (Clark Decl ¶ 94.) Martinez-Guerra expressly notes that one of ordinary skill in the art would understand that the boundaries between parser-translator subcomponents and their respective functionalities are variable and subject to compositions other than those described (Ex. 1002 at 9:26-31, 22:37-42), and one of skill in the art would appreciate that the dictionary entries may be part of the parser-translator. (Clark Decl. ¶ 94.)

As explained with reference to claim element 2B above, when the user submits the "delete /usr/extract/testing" command, the first input word "delete" is matched with a "regular token" entry in the dictionary having the same content ("delete"). (Clark Decl. ¶¶ 64, 95.) As Martinez-Guerra explains, a "regular token is a fixed sequence of characters (for example, the string 'delete')." (Ex. 1002 at 15:2-4.) The next (and final) input word the user submits is the pathname for the folder or file to be deleted ("/usr/extract/testing" in the example). (Clark Decl. ¶ 64, 95; Ex. 1002 at 15:57-16:5.) Martinez-Guerra explains that the dictionary entries are used to match the original input word with "pathname-expert." (Clark Decl. ¶¶ 64, 95; Ex. 1002 at 15:4-15, 15:63-16:5.) The parser-translator then uses the matching token "pathname-expert" instead of the original input word "/usr/extract/testing" when searching for a match in the command parse tree. (Clark Decl. ¶¶ 64, 95.)

8. [11A.2] "the parser configured for determining a presence of the matching token within the command parse tree for each input command word."

Martinez-Guerra discloses that the parser-translator determines a presence of the matching token within the command parse tree for each input word. In the example discussed above, the parser-translator first determines a presence of the token corresponding to the input word "delete" in the command parse tree, which reflects the initial parse state. (Clark Decl. ¶¶ 66, 96; Ex. 1002 at 15:50-62, 18:31-

48.) A matching token is found within the command parse tree because "delete" is the first token in the phrase structure rule RULE1. (Clark Decl. ¶¶ 66, 96; Ex. 1002 at 15:57; 20:40-45, Fig. 10.) Having matched the first token, the parser-translator then determines the presence of the second (and final) token, which Martinez-Guerra explains is "pathname-expert," corresponding to a valid file pathname supplied by the user, as explained above. (Clark Decl. ¶¶ 66, 96; Ex. 1002 at 15:50-16:5, 20:40-45.) The presence of "pathname-expert" is also found in the command parse tree because it corresponds to the final token of the phrase structure rule RULE1. (Clark Decl. ¶¶ 66, 96; Ex. 1002 at 15:57.) The parser-translator of Martinez-Guerra thus determines the presence of a matching token in the command parse tree for each input command word.

9. [12] "The system of claim 11, wherein the parser recursively traverses the command parse tree based on an order of the input command words for identification of the matching token within the identified one element."

Claim 12 recites the same method as claim 3, discussed above, but specifies that the method is performed by the parser of independent claim 10. As discussed with reference to claim 3, Martinez-Guerra teaches that the claimed method is performed by the disclosed parser-translator. (Clark Decl. ¶¶ 67-69, 97; Ex. 1002 at 20:31-34, 20:38-45.) Because the method performed by the parser of claim 12 is identical to that recited in claim 3, the analysis for this claim is the same as for

claim 3. Accordingly, Petitioner incorporates by reference and respectfully refers the Board to the discussion of claim 3, above.

10. [13] "The system of claim 12, wherein the parser validates at least a portion of the generic command by identifying the one element having the best match relative to the portion of the generic command."

As discussed above with reference to claim 10, the parser-translator validates the generic command "delete /usr/extract/testing" in the example given at column 15:50-62 because the token recognizer portion of the parser-translator determines, for each of the input words "delete" and "/usr/extract/testing" (where "/usr/extract/testing" is matched with the token "pathname-expert" in a dictionary entry), that the token corresponding to each input word is a next legal choice according to phrase structure rule RULE1. (Clark Decl. ¶¶ 89, 98.) By matching the final token ("pathname-expert") to the corresponding element in RULE1 ("pathname-expert"), Martinez-Guerra discloses that the parser-translator validates at least a portion of the generic command by identifying the one element having the best match relative to the portion of the generic command. (Clark Decl. ¶ 98; Ex. 1002 at 20:31-34.) Martinez-Guerra satisfies the limitation because validating the entire generic command "delete /usr/extract/testing" necessarily validates "at least a portion of" the generic command. (Clark Decl. ¶ 98.) Further, Martinez-Guerra discloses that if an input word is invalid or a command is incomplete—for example, if only the word "delete" is received without any pathname to define the

target of the operation—that action may result in returning an error. (See Ex. 1002 at 19:3-4.) In this circumstance, the parser-translator identifies a "best match" for less than the entire command. (Clark Decl. ¶ 98.)

C. Claim 14 and dependent claims 15-22 are invalid as obvious over Martinez-Guerra

1. [14A] "A computer readable medium having stored thereon sequences of instructions for executing a plurality of management programs according to respective command formats, the sequences of instructions including instructions for performing the steps of:"

Claim 14 recites the same method steps as claim 1, discussed above, but is drafted in the form of a computer-readable medium for performing the claimed steps. Martinez-Guerra discloses that the invention may be carried out by instructions embodied in a computer-readable medium. (Ex. 1002 at 4:52-56, 5:51-53, 13:55-14:7, 22:45-57, 24:14-34 (claims 16-17), 25:61-26:10 (claims 32-33).) Because the limitations of claim 14 following the preamble are identical to those of claim 1, claim 14 is obvious for the same reasons articulated as to claim 1. Accordingly, Petitioner incorporates by reference and respectfully refers the Board to the discussion of claim elements 1A-D above.

2. Dependent claims 15-19 and 21 are invalid as obvious over Martinez-Guerra

Dependent claims 15-19 and 21 recite the same limitations as dependent claims 2-6 and 8, respectively, but are drafted as dependent from computer-readable medium claim 14. As discussed above, Martinez-Guerra discloses that the

invention may be carried out by instructions embodied in a computer-readable medium. (*Id.* at 4:52-56, 5:51-53, 13:55-14:7, 22:45-57, 24:14-34 (claims 16-17), 25:61-26:10 (claims 32-33).) Because the limitations of dependent claims 15-19 and 21 are otherwise identical to dependent claims 2-6 and 8, claims 15-19 and 21 are obvious for the same reasons articulated as to claims 2-6 and 8, respectively. Accordingly, Petitioner incorporates by reference and respectfully refers the Board to the discussion of claims 2-6 and 8 above.

3. [20/22] "The medium of claim [19/21], further comprising instructions for performing the step of executing the prescribed command within the corresponding selected one management program."

Claims 20 and 22 recite the same limitation as claims 6 and 8, discussed above, but depend respectively from claims 19 and 21. Both claims 20 and 22 ultimately depend from computer-readable medium claim 14. As discussed with reference to claim 14, Martinez-Guerra discloses that the invention may be carried out by instructions embodied in a computer-readable medium. (Ex. 1002 at 4:52-56, 5:51-53, 13:55-14:7, 22:45-57, 24:14-34 (claims 16-17), 25:61-26:10 (claims 32-33).) Because claims 20 and 22 are otherwise identical to claims 7 and 9, the analysis for those claims applies equally to claims 20 and 22. Accordingly, Petitioner incorporates by reference and respectfully refers the Board to the discussion of claims 7 and 9 above.

D. Claim 23 and dependent claims 24-26 are invalid as obvious over Martinez-Guerra

1. [23A] "A system configured for executing a plurality of management programs according to respective command formats, the system comprising:"

The preamble of claim 23 recites the same system as claim 10, discussed above. Accordingly, Petitioner incorporates by reference and respectfully refers the Board to the discussion of claim element 10A above.

2. [23B.1] "means for validating a generic command received from a user,"

Martinez-Guerra discloses a parser-translator (*e.g.*, parser-translator 10 of Figure 1 and parser-translator 30 of Figure 3) that validates a generic command received from a user. (Clark Decl. ¶ 109.) As explained above with reference to claim element 10B.1, the discussion of which is incorporated by reference, Martinez-Guerra discloses that the parser-translator validates input words—including words for a generic command such as "delete /usr/extract/testing"—received from a user via an input stream. (*See* Clark Decl. ¶¶ 81-85; Ex. 1002 at Figs. 1, 3; 4:4-9; 9:15-23, 15:54-16:5.) The parser-translator of Martinez-Guerra thus includes the claimed "means for validating."

3. [23B.2] "the validating means configured for specifying valid generic commands relative to a prescribed generic command format and having elements each specifying at least one corresponding generic command component and a corresponding at least one command action value,"

Martinez-Guerra discloses that the parser-translator uses a grammar defining "a set of phrase structure rules and associated translation rules that define the syntactic order of a source and target language . . ." (Ex. 1002 at 7:57-59, *see also id.* at 16:15-16) and dictionary entries "that describes all the tokens that a parser-translator (*e.g.*, token recognizer 31 of parser-translator component 30) should recognize" (*id.* at 14:16-18.) As explained above with reference to claim element 10B.2, the discussion of which is incorporated by reference, the parser-translator is thus configured for specifying valid generic commands relative to a prescribed generic command format. (Clark Decl. ¶¶ 86-88, 110.)

As further discussed above with reference to claim element 10B.2, the "internal data structure[]" representation of the phrase structure rules created and maintained by the parser-translator component has elements specifying generic command components (*e.g.*, "delete," "pathname-expert") and a corresponding command action value (*e.g.*, returning an error or performing a translation according to the completed phrase structure rule's corresponding translation function) for each token of a phrase structure rule. (*See* Clark Decl. ¶ 87, 111; Ex. 1002 at 8:60-65, 14:11-12, 14:19-21, 15:50-62, 16:15-16, 17:47-58, 18:31-35,

19:3-4.) For these reasons and those specified above as to claim element 10B.2, Martinez-Guerra discloses this limitation.

4. [23B.3] "the validating means identifying one of the elements as a best match relative to the generic command; and"

As explained above with reference to claim element 10B.3, the discussion of which is incorporated by reference, Martinez-Guerra discloses that the parser-translator identifies the element corresponding to the final token needed to complete a phrase structure rule (*e.g.*, "pathname-expert") as the best match relative to the generic command (*e.g.*, "delete /usr/extract/testing"). (Clark Decl. ¶¶ 89, 112.) Because "pathname-expert" is the last token needed to satisfy the phrase structure rule "RULE1," it is the best match relative to the generic command "delete /usr/extract/testing." (*Id.*) For these reasons and those specified above as to claim 10B.3, Martinez-Guerra discloses this limitation.

5. [23C.1] "a plurality of translators configured for issuing commands for the management programs according to respective command formats,"

This limitation is identical to that of claim element 10C.1, discussed above, and the analysis is accordingly the same. Petitioner incorporates by reference and respectfully refers the Board to the discussion of claim element 10C.1 above.

6. [23C.2] "the validating means outputting a prescribed command to a selected one of the translators based on the identified one element."

This limitation is identical to that of claim element 10C.2, discussed above, except that claim element 23C.2 requires the claimed outputting to be performed by "the validating means" rather than "the parser," as in claim 10. As discussed above, the "validating means" includes the parser-translator and, accordingly, the analysis for this limitation is the same as claim element 10C.2. Petitioner incorporates by reference and respectfully refers the Board to the discussion of claim element 10C.2 above.

7. [24] "The system of claim 23, wherein the validating means comprises a command word translation table configured for storing for each prescribed command word a corresponding token for identification of a matching token, the validating means configured for determining a presence of the matching token for each input command word."

Claim 24 recites the same system as claim 11, discussed above, but requires the limitation be performed by "validating means" rather than the "parser," as in claim 11. As discussed in independent claim 23, the claimed "validating means" includes the same parser-translator discussed in claim 10 and, accordingly, the analysis for this claim is the same as for claim 11. Petitioner incorporates by reference and respectfully refers the Board to the discussion of claim 11 above.

8. [25] "The system of claim 24, wherein the validating means recursively validates each input command word based on an order of the input command words for identification of the matching token within the identified one element."

Claim 25 recites the same system as claim 12, discussed above, but requires the limitation be performed by "validating means" rather than the "parser," as in claim 12. As discussed in independent claim 23, the claimed "validating means" includes the same parser-translator discussed in claim 10 and, accordingly, the analysis for this claim is the same as for claim 12. Petitioner incorporates by reference and respectfully refers the Board to the discussion of claim 12 above.

9. [26] "The system of claim 25, wherein the validating means validates at least a portion of the generic command by identifying the one element having the best match relative to the portion of the generic command."

Claim 26 recites the same system as claim 13, discussed above, but requires the limitation be performed by the "validating means" rather than the "parser," as in claim 13. As discussed with respect to independent claim 23, however, the claimed "validating means" includes the same parser-translator discussed in claim 10 and, accordingly, the analysis for this claim is the same as for claim 13. Petitioner incorporates by reference and respectfully refers the Board to the discussion of claim 13 above.

IX. CONCLUSION

Petitioner respectfully requests institution of an *inter partes* review and cancellation of claims 1-26 of the '526 patent.

Respectfully submitted,

DATED: November 4, 2015

/s/ Eugene M. Paige

Eugene M. Paige, Registration No. 55,519
Counsel for Petitioner, Arista Networks, Inc.

Keker & Van Nest LLP
633 Battery Street
San Francisco, CA 94111
(415) 391-5400 (telephone); (415) 397-7188 (fax)

CERTIFICATION OF SERVICE (37 C.F.R. §§ 42.6(e), 42.105(a))

The undersigned hereby certifies that a copy of the foregoing **PETITION FOR *INTER PARTES* REVIEW OF U.S. PATENT NO. 7,047,526**, the accompanying Power of Attorney, list of exhibits for Petition for *Inter Partes* Review, and all associated exhibits were served in their entireties on the following parties via FedEx®:

Cisco c/o Leon R. Turkevich
Manelli Selter PLLC
10560 Main Street, Suite PH6
Fairfax, VA 22030
Address of record as listed on PAIR

Sean Sang-Chul Pak
John M. Neukom
Matthew Cannon
Quinn Emanuel Urquhart & Sullivan LLP
50 California Street, Floor 22
San Francisco, CA 94111
seanpak@quinnemanuel.com
johnneukom@quinnemanuel.com
Cisco-Arista@quinnemanuel.com
Additional address known to Petitioner as likely to effect service

Adam R. Alper
Kirkland & Ellis LLP
555 California Street
San Francisco, CA 94104
aalper@kirkland.com

Additional address known to Petitioner as likely to effect service

/s/ Eugene M. Paige
Eugene M. Paige, Registration No. 55,519
Counsel for Petitioner, Arista Networks, Inc.

Keker & Van Nest LLP
633 Battery Street
San Francisco, CA 94111
(415) 391-5400 (tel); (415) 397-7188 (fax)